



**TAKE  
CONTROL**  
[www.gdconf.com](http://www.gdconf.com)

MARCH 5-9  
**2007**  
SAN FRANCISCO

MOSCONE  
CENTER



## Next Gen: More Than Extra Channels?

⌚ Jason Page

⌚ Michael Kelly

⌚ SCEE – Technology Group



# Lets talk about ME!

- ⦿ Started in 1988 as a games programmer
  - Programmer (general “game” & audio)
  - Musician
    - ⦿ C64, Amiga, ST, Genesis, SNES, PC, PS one®, PlayStation®2 (PS2)..
      - ⦿ Audio Manager – SCEE Technology Group
        - Developer support
        - R&D



# Recent work

- ⦿ Last 3 years:
  - Designing the PLAYSTATION®3 (PS3™) MultiStream engine
  - Managing the PS3 MultiStream engine team



# What We're Going to Cover

- ⦿ Next Gen Audio:
  - What makes “Next Gen” next-gen?
  - What is realistically possible now?
  - What new skills do you need?
  - Potential problems



# How “Next Gen” is next-gen?

- ⊕ Can only speak from PS3 experience

Comparisons with PS2 audio?

- ⊕ Pong <> Half Life ?

Incredible potential

- ⊕ New working practices required.



# Audio channels

- ⦿ Think about audio creation..
  - C64 ... 3 channels. 4 waveforms. No DSP.
  - PS one ... 24 audio channels. One reverb unit
  - PS2 ... 48 audio channels
- ⦿ PC + Sequencer = “Unlimited” audio channels
  - Only limited to CPU
  - You don't think about the limitations.
- ⦿ \*\* At last, this is where we are now. \*\*





# Audio channels

- ⌚ Next Gen: “Unlimited” audio channels  
Of course there’s a limit
- ⌚ MultiStream is limited to 512 channels  
Leaves enough CPU for other purposes
  - ⌚ DSP effects
  - ⌚ Sub bus mixing / routing
  - ⌚ Codec processingRunning on 1 SPU (out of 6)





# Audio channels

- Will you ever need that many playing at once?
  - \*Probably not
  - But at last channel count is no longer an issue
- Artists have a palette of 16 million colours
  - It doesn't mean they have to use them all!

\*Don't quote me on that in 5 years time



But, as I said:

Next Gen is more than  
extra channels

[WWW.GDCONF.COM](http://WWW.GDCONF.COM)



# DSP effects

- ⌘ Governed by three things:
- ⌘ Used to be: Hardware capabilities
  - “Free” DSP power. No game resources required
- ⌘ Now: Processor power
  - Software DSP effects (VST plug-ins)
- ⌘ Most importantly: Portability
  - Easy editing / testing for audio engineers
  - Easy integration into game title.
  - Cross platform capabilities



# DSP effects

## ⌘ MultiStream:

- Impulse response reverb
- Pitch shifting / Time stretching
- Vocoder
- Multi-band compressor
- 3 band Parametric EQ
- 15 filter types, reverbs, delays, ring mod..
- And user can define their own

## ⌘ How powerful is the PS3?

We can run 50\*2 second impulses on a single SPU



# DSP effects

- ⊕ This is to audio what shaders are to graphics.
  - Far more scope to achieve goals
  - Far less boundaries
  - More than Delay / Reverb / Filter...
- ⊕ DSP effects don't meet your requirements?
  - Write your own!



# DSP effects

- ⦿ Emulation of other platform DSP effects.
  - Makes cross platform titles easier to create
  - ..And without having to aim for lowest common denominator



# Writing your own Next Gen audio engine.

- ⌚ LOTS to consider:
- ⌚ Simple things which you take for granted:
  - Variable re-sampling without clicks and pops
  - Amplitude Envelopes (ADSR)
  - “Basic” DSP effects (reverbs / filters)
  - Low latency





# Writing your own Next Gen audio engine.

- ⊕ As well as “Next Gen” things to learn:
  - Frequency domain processing (FFT / iFFT)
  - “Next Gen” DSP effects (Impulse response, pitch shift...)
  - Codecs (ATRAC3 / MP3 / OGG)
  - Licenses required for various formats
- ⊕ STILL has to be fast processing!
- ⊕ STILL has to require no memory!



# Next-Gen Implementation

How does this effect the audio engineer?

[WWW.GDCONF.COM](http://WWW.GDCONF.COM)



# The Old Way

- ⦿ Initialisation: Choose game reverb...  
“Hall”, “Studio” or “Cave” presets  
Set reverb depth  
(can’t modify in real time...CLICK! POP!)
- ⦿ Choose a speech sample...  
A robot? Under water? Talking through a radio?  
Add a bit of random pitch...  
Add a bit of random volume...



# The Next Gen approach

- ③ Which reverb to use in each “room”
  - Impulse responses for accurate reverb?
  - Time domain allows for far more real time control
    - ③ Modify parameters in real-time with no glitches
- ③ Use single speech sample
  - Set Compression / Ducking levels
  - Set Formant / Pitch shifting to help reduce repetition
  - Apply DSP for required effect:
    - ③ Vocoder (Robot)
    - ③ Filter (Underwater)
    - ③ Parametric EQ & Distortion (Radio)
- ③ Or apply them all
  - A robot, under water, on a radio..?!



# The Next Gen approach

- ⦿ Less need for pre-processing all sounds.
  - Affect audio in-game
  - More control. More flexibility
  - Less audio assets
  - More disk / RAM space for audio assets
  - Faster loading of levels
- ⦿ = Faster audio asset creation
  - There's more assets now, so don't get too excited



# What about music?

⌘ Currently:

Music streams:

- ⌘ Tracks streamed when required
- ⌘ Cross fading for interactivity
- ⌘ Takes a lot of disk space
- ⌘ Can cause latency issues (jumping between scores)
- ⌘ Interactivity problems (losing reverb when jumping scores)

⌘ BUT it sounds just like the score you wrote



# What about music?

- ⊕ Possible next-gen way:

MIDI (cue gasp from the audience):

- ⊕ Console plays MIDI file / audio data like PC did
- ⊕ All DSP's, samples and soft synths used on original track are processed in real time
- ⊕ Interactive (keeps reverb tail-offs as it's still processing reverb)
- ⊕ Faster to load (good for network titles)

- ⊕ Depends on Genre, but food for thought.

We're still not talking Giga-Sampler here...





# What about music?

- ⦿ And for streaming music:
- ⦿ 5.1 or 7.1 music scores
  - Surround sound is not just for SFX!
- ⦿ Care should be taken when folding down
  - Rear reverbs can cause phasing
  - Multiple mixes required?



# Sample Formats

- Sample heard a lot?
  - Enemy sounds (gun shots / footsteps...)
    - Many will be played at once, unlike player sounds
  - Keep it as PCM.
  - No CPU overhead for decoding data
- Commentary
  - Only 1 sentence being played at once
  - More likely to be compressed
  - Less disk streaming required
  - Less RAM required for buffering



# Format Choice

- ⦿ It's your choice.  
No need to use single standard formats  
Mix and match to get the best performance.
- ⦿ It is a decision for you to make.  
The audio job is changing.



# Routing

- ⊙ Decide on bus routing

Player specific bus?

- ⊙ Specific player DSP effects

Commentary bus

- ⊙ Audio bypasses room reverb
- ⊙ Audio sets ducking level for game

Room busses

- ⊙ Each room with its own reverb type
- ⊙ Each room routed to each other
- ⊙ All outputs affected by commentary amplitude

Master bus

- ⊙ Compression / final effects



# Audio Configuration

## ⦿ Fixed:

Same audio capabilities throughout game

- ⦿ Or all games from the studio

As with “hardware limited audio”

- ⦿ Easier to maintain
- ⦿ Easier to create tools for

Nothing extra to learn

- ⦿ Use VAG format for all sounds.
- ⦿ Just add reverb for when in a tunnel..

## ⦿ = Less Flexible

Does it sound “next-gen?”

Do the public think it’s worthy?



# Audio Configuration

- ⌵ Flexible:
  - Different audio capabilities throughout game.
- ⌵ More channels during level than boss level?
  - ⌵ More DSP power available in boss level
  - ⌵ Different routing required
  - ⌵ Make the most out of CPU and RAM
  - ⌵ More resources to manage
  - ⌵ Far more flexible
- ⌵ BETTER SOUNDING GAMES



# What does this do for content creation?

## ⚙️ Helps:

Things sound just as you want them to.  
Each job requires a different approach  
Keeps things fresh and original

## ⚙️ Hinders:

No standard work practice?  
Everything has different approach and specific tools  
Difficult – especially for freelance engineers





# Musicians new role?

## ⌚ Decide on required audio capabilities:

Allocation of channels

Audio routing

- ⌚ MultiStream has 31 REAL sub busses
- ⌚ Each with 8 DSP slots
- ⌚ Independent volume controls
- ⌚ Routing to any other busses

Use of available DSP effects

- ⌚ Decide if new DSP effects need creating

Format of sample data



# Musicians new role?

- ⊕ Work closer with programmers / producers / designers

As ever...Don't we say this every year?!

More shared assets than before.

More need for programmers to be involved

More need to decide on memory and RAM usage.

- ⊕ Freelancers NEED to be aware of this too.



# Which brings us onto tools and working practices...

- ④ SCEE's approach to help solve these issues

Over to Michael:

[WWW.GDCONF.COM](http://WWW.GDCONF.COM)



# Designing Tools for Next-Gen Audio Engines

[WWW.GDCONF.COM](http://WWW.GDCONF.COM)



# Why new tools?

- ④ Next-gen engines offer greater complexity
  - Tools need to be more flexible
- ④ Interactive Audio Tools are still in their infancy (or perhaps teenagers)
  - 50% of these slides are not even about next-gen



# What are audio tools for

## GUI Tools

Designed to separate creative design from low-level technical detail

Put control of audio in Sound Designer's hands

Ease communicate between Programmer and Sound Designer



# Previous-gen tools

- ⌚ Designed around audio engine
  - Expose what's in engine
- ⌚ Linked to audio hardware
  - Reverb, ADSR, Volume, Pitch
- ⌚ Uses hardware limitations
  - Voice limits, bus grouping
- ⌚ Same as being linked to a single audio engine



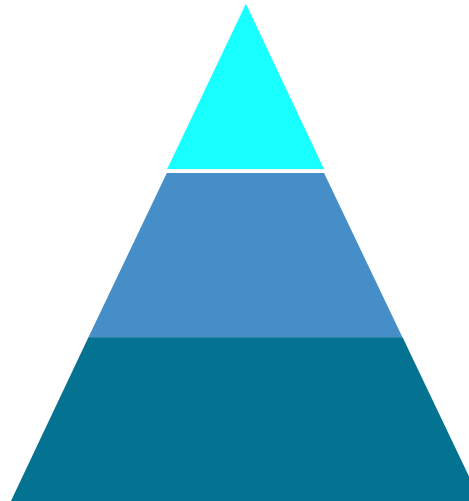


# Platform / tool relationship

Tool

API

Platform



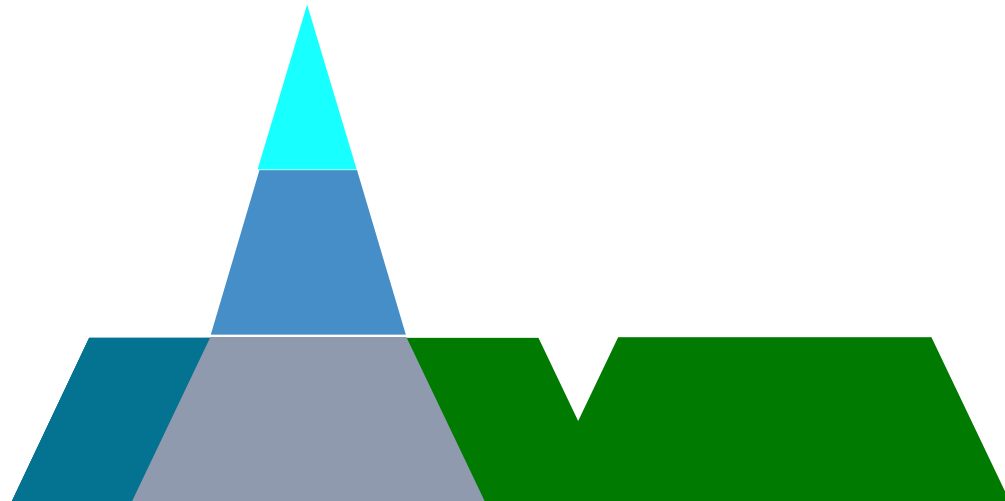


# Cross-platform tools

Tool

API

Platform



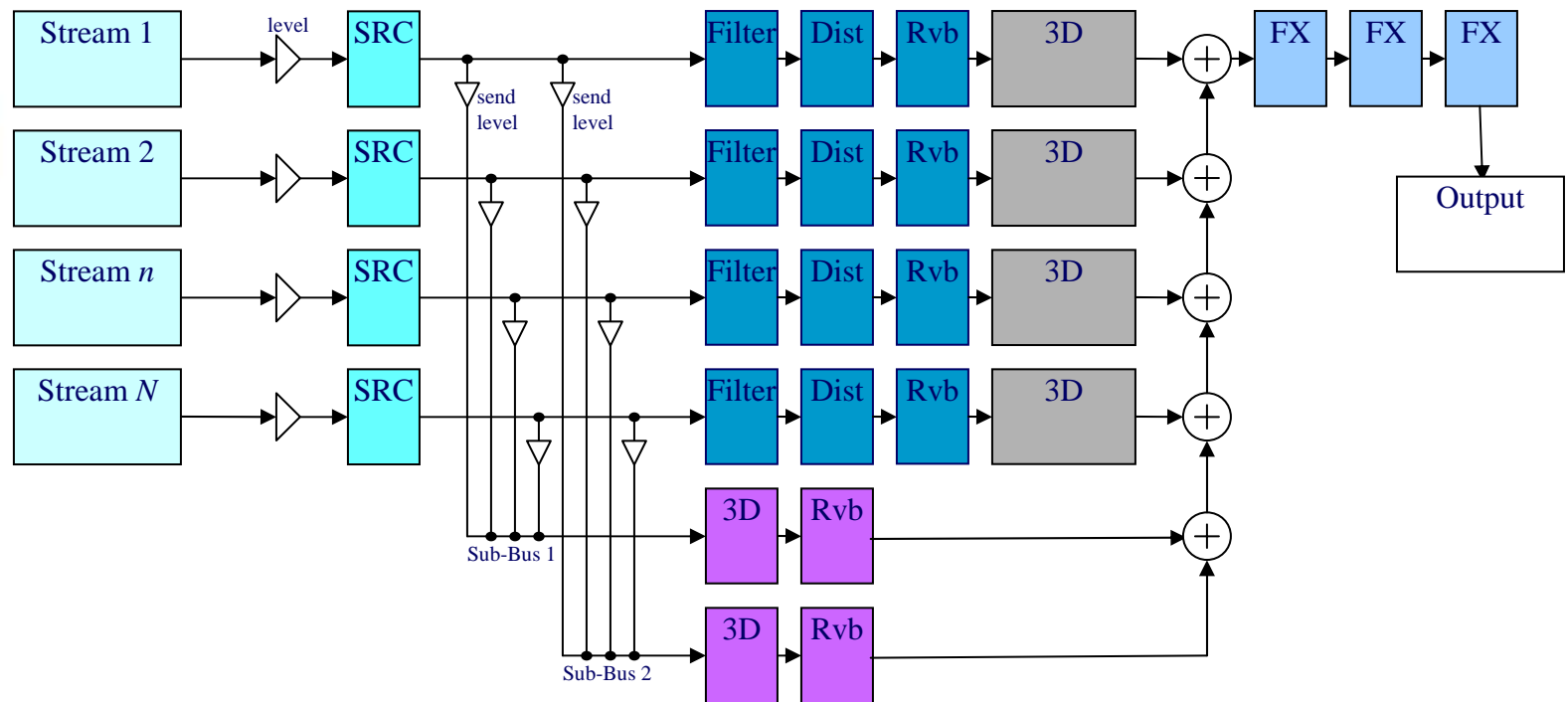


# What changed

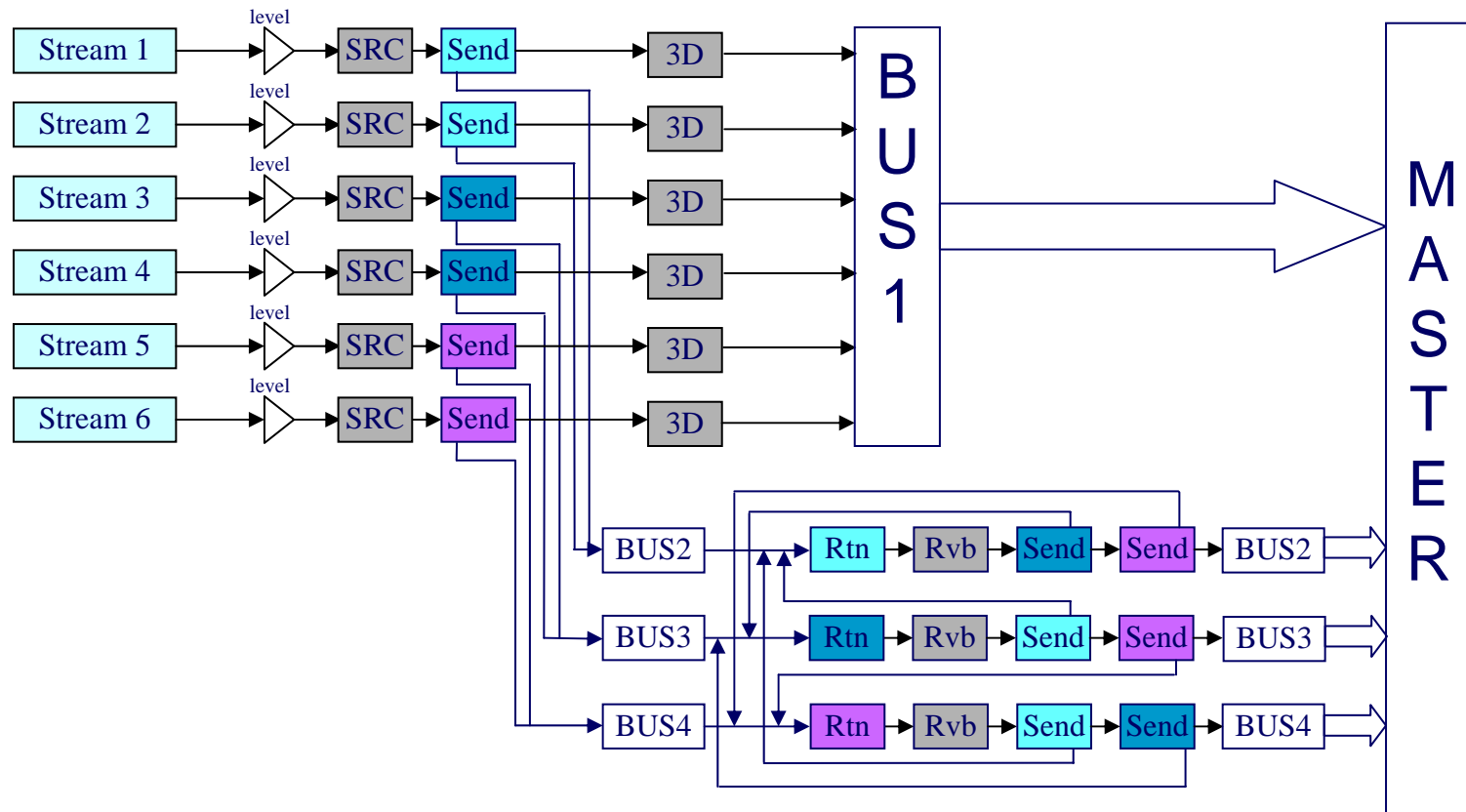
- ⌚ With next-gen there's still previous-gen  
SCE still has PS2 and PSP
- ⌚ Hardware engines aren't the only way
- ⌚ Software engines are more flexible  
Tool design is more complex  
-can't tie a tool to one config
- ⌚ Next-gen isn't just audio!



# Routing A



# Routing B





# Next-gen isn't just audio

- ⌚ There are physics engines and other things too!
- ⌚ Footsteps case study (previous gen)
  - ⌚ Artist creates animation of person walking
  - ⌚ Anim frames tagged with footstep triggers
  - ⌚ Game randomises footstep samples



# Next-gen isn't just audio

- ④ Table dropping case study (next-gen)
  - ④ Artist creates a table object
  - ④ Physics engine calculates table trajectory
  - ④ Audio programmer uses thresholds to find table hitting object
  - ④ Does the audio get triggered every time the threshold is crossed?
    - ④ - It does in one game!
    - ④ - Sounds like a machine gun



# Physics / collision problems

- ④ Can be solved by clever programming
- ④ How can this be addressed in a tool

Greater access to real time parameter

Requires greater collaboration between programmer and sound designer?

④ - NO (ish)





# How do you approach this?

## ⊕ SCEE approach:

### Core Engine

- ⊕ MultiStream API used by programmer

### Audio config tool

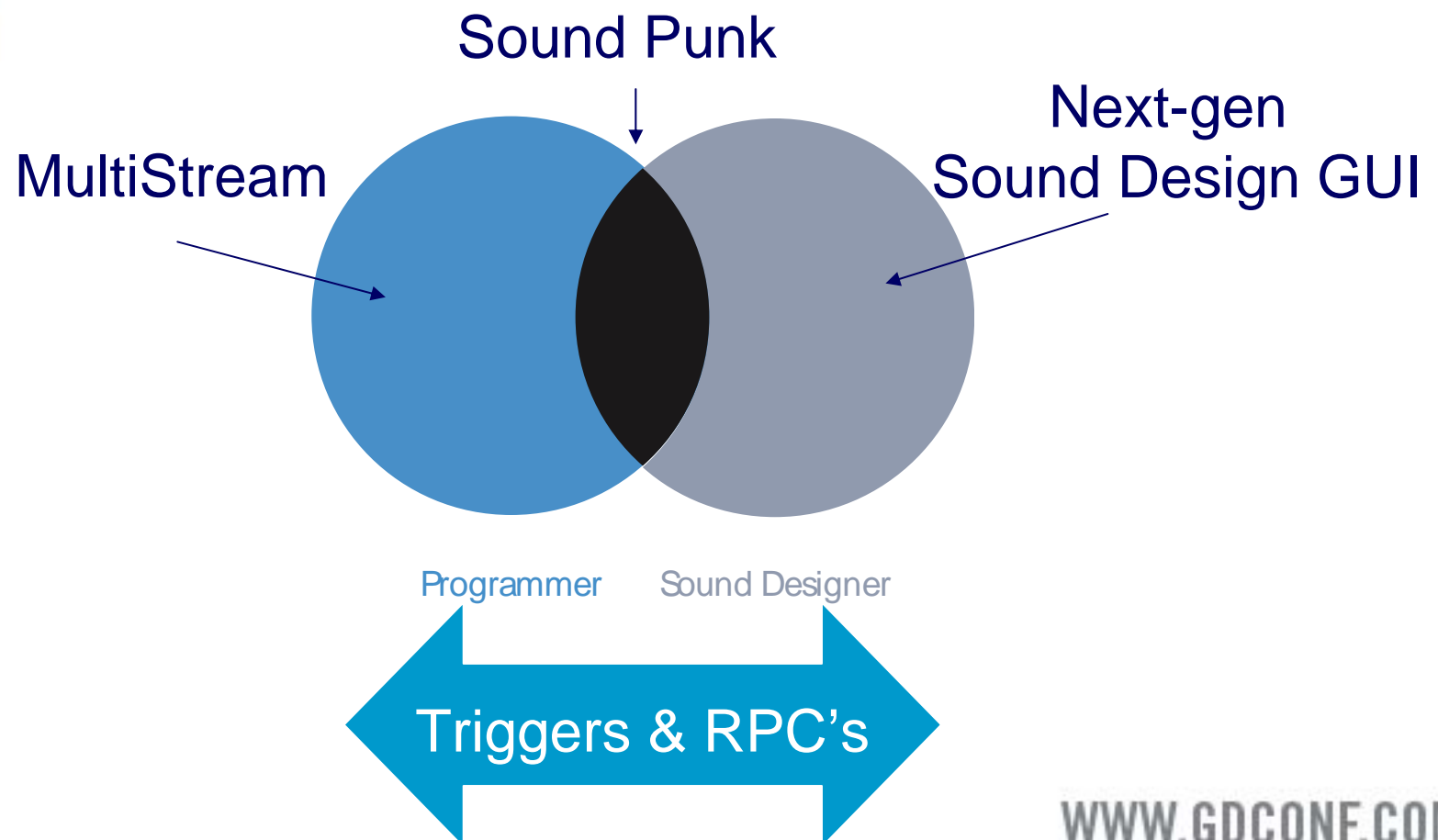
- ⊕ Sound Punk configures MultiStream
  - ⊕ Configuration can be shared with sound design
- ### GUI Tools

### Sound Design GUI Tools

- ⊕ In development



# Sound design workflow





# Tool specification

- ⊙ Configurable
- ⊙ Plug-able (audio)
- ⊙ Separate from runtime
- ⊙ Data driven
- ⊙ Open interface
  - Extensible via plugins
- ⊙ Offline and runtime processing
- ⊙ Assets from anywhere – static, streaming, network, user generated



# MultiStream VST Plugins





# Sound design tool design

- ④ Asset handling
  - PCM and sequence data + platform specific
- ④ Right tool for the right job
  - Content creation
    - ④ Sound Forge, Audition, ProTools etc.
  - Sequencing
    - ④ Cubase, logic, etc.
  - Asset management
    - ④ Source Safe, CVS, Perforce, etc.



# Integration and interoperability

- ④ Work with existing tools
- ④ Work with in-game and external controllers
- ④ Work with standard formats
- ④ Extensible via plugins (VSTs)
  - Must also be supported in API/config tool
- ④ Open file format
  - iXMF and other standards?
    - ④ Still waiting
    - ④ Could cover user or online generated assets too





# Tool challenges

- ④ Multiplatform tools are fine
  - how do you really do cross platform and get the best from each one?
- ④ Standardizing plugin support?
- ④ Scripting flexibility without making sound designers into programmers
  - Complex scripting / intuitive interface
- ④ What should a sound design tool look like?



# The big picture

- ⌘ Getting this right is good for our industry
- ⌘ There is no accepted interface for a sound design tool
- ⌘ iXMF is a good start
  - ⌘ Draws attention to tool design
  - ⌘ Encourages interchangeability
  - ⌘ Helps separate tools from engines
- ⌘ Better recruits
- ⌘ More informed sound designers and programmers
  - Not constantly re-inventing the wheel
- ⌘ Better use of specific audio engine features





# To summarise

- ③ This time next-gen **IS** different  
It's not just more channels
- ③ Ignore this  
Games will sound like old games  
More importantly, they'll sound worse than your competitors'!
- ③ Embrace it  
And your games will sound amazing
- ③ Next-gen needs good tools
- ③ Consider flexible tool design  
Middleware & in-house



# Where to start

- ③ Take a fresh look at your game /design
  - ③ Imagine you can do ANYTHING
  - ③ Make a wish list – routing, DSP, effects
- ③ Look at the PS3 MultiStream SDK
  - ③ (sound designers and programmers)
  - ③ The effects and routing are probably already there – they were our wish list
  - ③ If not – write your own plugins!
- ③ Think about tools early on
  - ③ Not just what you need for current project



# Thank You

[WWW.GDCONF.COM](http://WWW.GDCONF.COM)